UNITED STATES PATENT AND TRADEMARK OFFICE

A

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/667,122	09/21/2000	William C. Moyer	SC11306TH	9170

23125 7590 10/28/2005

FREESCALE SEMICONDUCTOR, INC.
LAW DEPARTMENT
7700 WEST PARMER LANE MD:TX32/PL02
AUSTIN, TX 78729

EXAMINER	
HUISMAN, DAVID J	

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 10/28/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/667,122	MOYER ET AL.	
	Examiner	Art Unit	
	David J. Huisman	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *16 August 2005*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-13,15-21 and 23-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-13,15-21 and 23-27* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *21 September 2000* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-13, 15-21, and 23-27 have been examined.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: RCE and Amendment as received on 8/16/2005.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claims 1-9, 13, 15-17, 21, 23, and 25 are rejected under 35 U.S.C. 102(b) as being anticipated by Hoyt et al., U.S. Patent No. 5,604,877 (as applied in the previous Office Action and herein referred to as Hoyt). In addition, Hennessy and Patterson, "Computer Architecture - A Quantitative Approach, 2nd Edition," 1996 (herein referred to as Hennessy), is cited as extrinsic evidence for providing a showing that prediction signals and predicted addresses are provided with each other in the same cycle.

5. Referring to claim 1, and according to a first interpretation, Hoyt has taught a processing system for accessing memory, comprising:

a) an address bus for providing a current address and a previous address to memory, wherein the current address follows the previous address without any intervening addresses. See Fig.3 and note the fetch unit is coupled to memory via address bus. An address is transmitted on the bus and applied to the memory in order to fetch instructions. Also, it is inherent that the system will provide a current address and a previous address with no intervening addresses. This occurs when the current address is provided immediately after a previous address (i.e., in a row).

b) a data bus for receiving information from memory. See column 5, lines 11-19, and note that information is loaded from and stored to cache/memory. The information inherently is transmitted via data bus.

c) generating a first sequence signal that when negated indicates that the current address may not be sequential to the previous address. See column 8, lines 41-58, and note that the previous address is applied to the Branch Target Buffer Circuit 40 (BTB). Note that if the previous address corresponds to a branch instruction, a "hit" will occur in the BTB, and a prediction will be provided. This "hit" is the first signal because when negated, it indicates that a branch has been located within the BTB, and consequently, the current address (next address from which to fetch) may not be sequential to the previous address because a branch instruction could result in the next address being sequential (when not taken) or not being sequential (when taken).

d) generating a second sequence signal that when negated indicates that the current address is not sequential to the previous address. See Table 2 and column 9, lines 59-62, and note that the second sequence signal would be the predicted direction of the branch. If the branch is predicted taken, then at that point in time, the current address from which instructions are fetched is not sequential to the previous address.

e) generating a third sequence signal that when negated indicates that the current address, if it is an instruction address, is not sequential to the previous address that was an instruction address. See column 2, lines 3-9, and note that the third signal would be the branch outcome signal, which determines if a misprediction has occurred or not. That is, if the branch was predicted taken, for instance, when the third signal is negated (signifying correct prediction), then the current address, which is an instruction address, is not sequential. On the other hand, if the third signal is asserted (signifying incorrect prediction), then the branch was really not supposed to be taken, and the current address is sequential to the previous address. Note that the examiner is defining the previous address as the branch instruction address, and the current address is the address from which to fetch after the previous address.

6. Referring to claim 1, and according to a second interpretation, Hoyt has taught a processing system for accessing memory, comprising:

a) an address bus for providing a current address and a previous address to memory, wherein the current address follows the previous address without any intervening addresses. See Fig.3 and note the fetch unit is coupled to memory via address bus. An address is transmitted on the bus and applied to the memory in order to fetch instructions. Also, it is inherent that the system will provide a current address and a previous address with no intervening addresses. This occurs when the current address is provided immediately after a previous address (i.e., in a row).

b) a data bus for receiving information from memory. See column 5, lines 11-19, and note that information is loaded from and stored to cache/memory. The information inherently is transmitted via data bus.

c) generating a first sequence signal that when negated indicates that the current address may not be sequential to the previous address. See column 8, lines 41-58, and note that the previous address is applied to the Branch Target Buffer Circuit 40 (BTB). Note that if the previous address corresponds to a branch instruction, a "hit" will occur in the BTB, and a prediction will be provided. This "hit" is the first signal because when negated, it indicates that a branch has been located within the BTB, and consequently, the current address (next address from which to fetch) may not be sequential to the previous address because a branch instruction could result in the next address being sequential (when not taken) or not being sequential (when taken).

d) generating a second sequence signal that when negated indicates that the current address is not sequential to the previous address. The second signal would correspond to the branch outcome of the branch, i.e., whether or not the branch was taken. If the branch turns out to be taken then the signal indicates that the current address is not sequential to the previous address. This signal will affirm a taken prediction.

e) generating a third sequence signal that when negated indicates that the current address, if it is an instruction address, is not sequential to the previous address that was an instruction address. In Hoyt, the third signal would correspond to the signal that causes the program counter (PC) to either be incremented to point to the next sequential instruction or to be replaced with a branch target address. Processors inherently include a PC, which is used to point to the next instruction in memory to be fetched. When no branches occur (or not-taken branches occur), the PC is simply incremented, thereby indicating that sequential instructions at sequential addresses are fetched. The PC can either be incremented or replaced with a branch target. When the third signal is negated (don't increment and replace PC with branch target), the current address is not

sequential to the preceding address. When the third signal is asserted (increment PC), the current address is sequential.

7. Referring to claim 2, and according to the first interpretation, Hoyt has taught a processing unit as described in claim 1. Hoyt has further taught that if the current address is not sequential to the previous address, the first sequence signal is negated prior to the second sequence signal being negated. Clearly, the branch must be located within the table before a prediction can be provided. Consequently, the first signal (hit signal) would be negated before the second signal (prediction signal) is negated.

8. Referring to claim 2, and according to the second interpretation, Hoyt has taught a processing unit as described in claim 1. Hoyt has further taught that if the current address is not sequential to the previous address, the first sequence signal is negated prior to the second sequence signal being negated. Clearly, the branch must be located within the table and predicted before the outcome is known. Consequently, the first signal (hit signal) would be negated before the second signal (outcome signal) is negated.

9. Referring to claim 3, and according to a first interpretation, Hoyt has taught a processing system for accessing memory, comprising:

a) an address bus for providing a current address and a previous address to memory, wherein the current address follows the previous address without any intervening addresses. See Fig.3 and note the fetch unit is coupled to memory via address bus. An address is transmitted on the bus and applied to the memory in order to fetch instructions. Also, it is inherent that the system will provide a current address and a previous address with no intervening addresses. This occurs when the current address is provided immediately after a previous address (i.e., in a row).

b) a data bus for receiving information from memory. See column 5, lines 11-19, and note that information is loaded from and stored to cache/memory. The information inherently is transmitted via data bus.

c) an execution unit which generates branch conditions and data addresses. See column 1, lines 44-47, and note that branch conditions are resolved in an execution stage. Therefore, it is inherent that an execution unit exists to execute the branch instruction. Furthermore, it is further inherent that if a system loads and stores data to memory, as in Hoyt, then an execution unit must exists to execute load and store instructions such that data addresses for these operations are provided.

d) a decode control unit which decodes instructions. See Fig.3, component 60.

c) a fetch unit (Fig.3, components 30 and 35), coupled to the execution unit, the decode control unit, the address bus, and the data bus, for generating a first sequence signal that when negated indicates that the current address may not be sequential to the previous address. See column 8, lines 41-58, and note that the previous address is applied to the Branch Target Buffer Circuit 40 (BTB). Note that if the previous address corresponds to a branch instruction, a "hit" will occur in the BTB, and a prediction will be provided. This "hit" is the first signal because when negated, it indicates that a branch has been located within the BTB, and consequently, the current address (next address from which to fetch) may not be sequential to the previous address because a branch instruction could result in the next address being sequential (when not taken) or not being sequential (when taken).

d) generating a second sequence signal that when negated indicates that the current address is not sequential to the previous address. See Table 2 and column 9, lines 59-62, and note that the

second sequence signal would be the predicted direction of the branch. If the branch is predicted taken, then at that point in time, the current address from which instructions are fetched is not sequential to the previous address.

e) generating a third sequence signal that when negated indicates that the current address, if it is an instruction address, is not sequential to the previous address that was an instruction address. See column 2, lines 3-9, and note that the third signal would be the branch outcome signal, which determines if a misprediction has occurred or not. That is, if the branch was predicted taken, for instance, when the third signal is negated (signifying correct prediction), then the current address, which is an instruction address, is not sequential. On the other hand, if the third signal is asserted (signifying incorrect prediction), then the branch was really not supposed to be taken, and the current address is sequential to the previous address. Note that the examiner is defining the previous address as the branch instruction address, and the current address is the address from which to fetch after the previous address.

10. Referring to claim 3, and according to a second interpretation, Hoyt has taught a processing system for accessing memory, comprising:

a) an address bus for providing a current address and a previous address to memory, wherein the current address follows the previous address without any intervening addresses. See Fig.3 and note the fetch unit is coupled to memory via address bus. An address is transmitted on the bus and applied to the memory in order to fetch instructions. Also, it is inherent that the system will provide a current address and a previous address with no intervening addresses. This occurs when the current address is provided immediately after a previous address (i.e., in a row).

b) a data bus for receiving information from memory. See column 5, lines 11-19, and note that information is loaded from and stored to cache/memory. The information inherently is transmitted via data bus.

c) an execution unit which generates branch conditions and data addresses. See column 1, lines 44-47, and note that branch conditions are resolved in an execution stage. Therefore, it is inherent that an execution unit exists to execute the branch instruction. Furthermore, it is further inherent that if a system loads and stores data to memory, as in Hoyt, then an execution unit must exists to execute load and store instructions such that data addresses for these operations are provided.

d) a decode control unit which decodes instructions. See Fig.3, component 60.

c) a fetch unit (Fig.3, components 30 and 35), coupled to the execution unit, the decode control unit, the address bus, and the data bus, for generating a first sequence signal that when negated indicates that the current address may not be sequential to the previous address. See column 8, lines 41-58, and note that the previous address is applied to the Branch Target Buffer Circuit 40 (BTB). Note that if the previous address corresponds to a branch instruction, a "hit" will occur in the BTB, and a prediction will be provided. This "hit" is the first signal because when negated, it indicates that a branch has been located within the BTB, and consequently, the current address (next address from which to fetch) may not be sequential to the previous address because a branch instruction could result in the next address being sequential (when not taken) or not being sequential (when taken).

d) generating a second sequence signal that when negated indicates that the current address is not sequential to the previous address. The second signal would correspond to the branch outcome

of the branch, i.e., whether or not the branch was taken. If the branch turns out to be taken then the signal indicates that the current address is not sequential to the previous address. This signal will affirm a taken prediction.

e) generating a third sequence signal that when negated indicates that the current address, if it is an instruction address, is not sequential to the previous address that was an instruction address. In Hoyt, the third signal would correspond to the signal that causes the program counter (PC) to either be incremented to point to the next sequential instruction or to be replaced with a branch target address. Processors inherently include a PC, which is used to point to the next instruction in memory to be fetched. When no branches occur (or not-taken branches occur), the PC is simply incremented, thereby indicating that sequential instructions at sequential addresses are fetched. The PC can either be incremented or replaced with a branch target. When the third signal is negated (don't increment and replace PC with branch target), the current address is not sequential to the preceding address. When the third signal is asserted (increment PC), the current address is sequential.

11. Referring to claim 4, Hoyt has taught a system as described in claim 3. Furthermore, it is inherent that Hoyt's decode control unit comprises an instruction register (IR). An IR is a known component which holds the instruction that is to be decoded and executed.

12. Referring to claim 5, Hoyt has taught a system as described in claim 3. Hoyt has further taught an address control unit, coupled to the decode control unit and the execution unit, for receiving a branch condition signal and a branch decode signal and a load/store signal and for providing the first, second, and third sequence signals. See Fig.3 ands note that the address control unit may comprise any of the components 30, 40, 50, 70, and 80. These components are

coupled to execution units 90, and decoder 60. All of the claimed signals are produced by these components (note the connections of components in Fig.3).

13. Referring to claim 6, Hoyt has taught a system as described in claim 5. Hoyt has further taught that the execution unit comprises a condition generator that provides the branch condition signal. See column 1, lines 44-47.

14. Referring to claim 7, Hoyt has taught a system as described in claim 6. Hoyt has further taught that the execution unit comprises a data address generator which provides a data address signal to the fetch unit. Clearly, if an execution unit is performing load instructions, then a data address signal must be provided to a fetch unit.

15. Referring to claim 8, and according to the first interpretation, Hoyt has taught a system as described in claim 7. Hoyt has further taught that if the current address is not sequential to the previous address, the first sequence signal is negated prior to the second sequence signal being negated. Clearly, the branch must be located within the table before a prediction can be provided. Consequently, the first signal (hit signal) would be negated before the second signal (prediction signal) is negated.

16. Referring to claim 8, and according to the second interpretation, Hoyt has taught a system as described in claim 7. Hoyt has further taught that if the current address is not sequential to the previous address, the first sequence signal is negated prior to the second sequence signal being negated. Clearly, the branch must be located within the table and predicted before the outcome is known. Consequently, the first signal (hit signal) would be negated before the second signal (outcome signal) is negated.

17. Referring to claim 9, and according to the first interpretation, Hoyt has taught a system as described in claim 3. Hoyt has further taught that if the current address is not sequential to the previous address, the first sequence signal is negated prior to the second sequence signal being negated. Clearly, the branch must be located within the table before a prediction can be provided. Consequently, the first signal (hit signal) would be negated before the second signal (prediction signal) is negated.

18. Referring to claim 9, and according to the second interpretation, Hoyt has taught a system as described in claim 3. Hoyt has further taught that if the current address is not sequential to the previous address, the first sequence signal is negated prior to the second sequence signal being negated. Clearly, the branch must be located within the table and predicted before the outcome is known. Consequently, the first signal (hit signal) would be negated before the second signal (outcome signal) is negated.

19. Referring to claim 13, Hoyt has taught a processor system comprising:

a) an execution unit. See Fig.3, component 90.

b) a decode control unit. See Fig.3, component 60.

c) a fetch unit, coupled to the execution unit and the decode control unit, for providing addresses on an address bus which may be sequential and providing a first sequence signal and a second sequence signal for each address provided on the address bus wherein the first sequence signal indicates whether each address provided on the address bus may be sequential to an immediately preceding address on the bus and the second signal indicates whether each address provided on the bus is sequential to the immediately preceding address on the bus. It should be noted from Hoyt's system that instructions are continuously fetched, and these instructions may include a

branch instruction. Consequently, the current instruction address will inherently be sequential to the previous instruction address unless a taken branch instruction is encountered. Therefore, the first signal would be the BTB hit signal, as a hit denotes that a branch has been encountered and that the current address (address after the branch address) may be sequential to the previous address. Note that because the current address may only be predicted, whether or not it is sequential is not yet known. See column 8, lines 42-53. In addition, a second signal would correspond to the branch outcome. This second signal would be a signal that indicates whether the current address is sequential to the preceding address. That is, when a branch has been taken, if the prediction was taken, the second signal will indicate that the prediction was correct and that the current address is not sequential to the preceding address. If the prediction was taken, but the branch was not taken, then the second signal would indicate that the prediction was wrong and that the current address is sequential to the preceding address.

d) wherein if the second sequence signal corresponding to one of the addresses indicates that the address is not sequential to the immediately preceding address (i.e., the branch is taken):

d1) the first sequence signal corresponding to the address indicates that the address may not be sequential to the immediately preceding address prior to the second sequence signal indicating that the address is not sequential to the immediately preceding address. Since the first signal is associated with predicting a branch and the second signal is associated with resolving a branch, the first signal clearly indicates prior to the second signal indicating.

d2) the second sequence signal indicates that the address is not sequential to the immediately preceding address in response to resolving a conditional branch. As described

above, the second signal is not determined until the true branch outcome is known (in response to the branch being resolved).

20. Referring to claim 15, Hoyt has taught a system as described in claim 13. Hoyt has further taught that the addresses may be instruction addresses, and wherein the fetch unit further provides a third sequential signal which indicates whether each address that is an instruction address is sequential to a previous instruction address. In Hoyt, the third signal would correspond to the signal that causes the program counter (PC) to be incremented to point to the next sequential instruction. Processors inherently include a PC, which is used to point to the next instruction in memory to be fetched. When no branches occur (or not-taken branches occur), the PC is simply incremented, thereby indicating that sequential instructions at sequential addresses are fetched. The PC can either be incremented or replaced with a branch target. In the former case, the signal that causes the PC to be incremented is the third signal.

21. Referring to claim 16, Hoyt has taught a system as described in claim 15. Hoyt has further taught that the execution unit comprises a condition generator that provides a branch condition signal to the fetch unit. See column 1, lines 44-47.

22. Referring to claim 17, Hoyt has taught a system as described in claim 16. Hoyt has further taught that the decode control unit provides a branch decode signal and a load/store signal to the fetch unit. Clearly, all instructions are decoded, and in response to decoding a branch instruction, decoder 60 (Fig.3) will provide a branch decode signal. In addition, since loads/stores do occur in Hoyt (see column 5, lines 11-19), the load/store signals must be generated.

23. Referring to claim 21, and according to the second interpretation, Hoyt has taught a processing unit as described in claim 2. Hoyt has further taught that the second sequence signal is negated in response to resolving a conditional branch. As described above, the second signal would correspond to the branch outcome of the branch. Therefore, the signal is determined after resolving the branch.

24. Referring to claim 23, and according to the first interpretation, Hoyt has taught a processing unit as described in claim 2. Hoyt has further taught that if the current address is not sequential to the previous address, the first sequence signal is negated prior to the second sequence signal being negated and the first and second sequence signals are negated in a same clock cycle. It should be noted that the first sequence signal is always negated before the second sequence signal because there must be a hit (first signal) in the BTB before a prediction (second signal) can be provided. In addition, it is known that these signals occur in the same cycle so that a predicted address may be outputted for fetching purposes in the very next cycle, thereby keeping the pipeline full. See column 1, lines 38-41. That is, if a branch instruction is fetched in clock cycle X, then a hit will occur and a prediction will be made in that cycle, so that in clock cycle X+1, the predicted path may be fetched. The first and second signals are negated in the same cycle in which the current address is provided because the predicted address is part of the prediction, and the current address is used in the very next cycle to fetch from (however, it is generated in the same cycle in which the previous address is applied to the predictor). This is done to keep the pipeline full, and is further supported by pages 271-273 of Hennessy. That is, if a hit occurs in a branch buffer, then a prediction and a current address (in this case, the predicted

address) are provided at the same time. See Fig.4.22 (note that the prediction and address are read from the buffer at the same time in response to a hit of the previous address).

25. Referring to claim 24, Hoyt has taught a processing system for accessing memory, comprising:

a) an address bus for providing a current address and a previous address to memory. See Fig.3 and note the fetch unit is coupled to memory via address bus. An address is transmitted on the bus and applied to the memory in order to fetch instructions. Also, it is inherent that the system will provide a current address and a previous address.

b) a data bus for receiving information from memory. See column 5, lines 11-19, and note that information is loaded from and stored to cache/memory. The information inherently is transmitted via data bus.

c) an execution unit which generates branch conditions and data addresses. See column 1, lines 44-47, and note that branch conditions are resolved in an execution stage. Therefore, it is inherent that an execution unit exists to execute the branch instruction. Furthermore, it is further inherent that if a system loads and stores data to memory, as in Hoyt, then an execution unit must exists to execute load and store instructions such that data addresses for these operations are provided.

d) a decode control unit which decodes instructions. See Fig.3, component 60.

c) a fetch unit (Fig.3, components 30 and 35), coupled to the execution unit, the decode control unit, the address bus, and the data bus, for generating a first sequence signal that when negated indicates that the current address may not be sequential to the previous address. See column 8, lines 41-58, and note that the previous address is applied to the Branch Target Buffer Circuit 40

(BTB). Note that if the previous address corresponds to a branch instruction, a "hit" will occur in the BTB, and a prediction will be provided. This "hit" is the first signal because when negated, it indicates that a branch has been located within the BTB, and consequently, the current address (next address from which to fetch) may not be sequential to the previous address because a branch instruction could result in the next address being sequential (when not taken) or not being sequential (when taken).

d) generating a second sequence signal that when negated indicates that the current address is not sequential to the previous address. See Table 2 and column 9, lines 59-62, and note that the second sequence signal would be the predicted direction of the branch. If the branch is predicted taken, then at that point in time, the current address from which instructions are fetched is not sequential to the previous address.

e) generating a third sequence signal that when negated indicates that the current address, if it is an instruction address, is not sequential to the previous address that was an instruction address. See column 2, lines 3-9, and note that the third signal would be the branch outcome signal, which determines if a misprediction has occurred or not. That is, if the branch was predicted taken, for instance, when the third signal is negated (signifying correct prediction), then the current address, which is an instruction address, is not sequential. On the other hand, if the third signal is asserted (signifying incorrect prediction), then the branch was really not supposed to be taken, and the current address is sequential to the previous address. Note that the examiner is defining the previous address as the branch instruction address, and the current address is the address from which to fetch after the previous address.

f) wherein if the current address is not sequential to the previous address, the first sequence signal is negated prior to the second sequence signal being negated and the first and second sequence signals are negated in a same clock cycle during which the current address is provided. It should be noted that the first sequence signal is always negated before the second sequence signal because there must be a hit (first signal) in the BTB before a prediction (second signal) can be provided. In addition, it is known that these signals occur in the same cycle so that a predicted address may be outputted for fetching purposes in the very next cycle, thereby keeping the pipeline full. See column 1, lines 38-41. That is, if a branch instruction is fetched in clock cycle X, then a hit will occur and a prediction will be made in that cycle, so that in clock cycle X+1, the predicted path may be fetched. The first and second signals are negated in the same cycle in which the current address is provided because the predicted address is part of the prediction, and the current address is used in the very next cycle to fetch from (however, it is generated in the same cycle in which the previous address is applied to the predictor). This is done to keep the pipeline full, and is further supported by pages 271-273 of Hennessy. That is, if a hit occurs in a branch buffer, then a prediction and a current address (in this case, the predicted address) are provided at the same time. See Fig.4.22 (note that the prediction and address are read from the buffer at the same time in response to a hit of the previous address).

26. Referring to claim 25, Hoyt has taught a system as described in claim 13. Hoyt has further taught that if the second sequence signal indicates that an address is not sequential to the immediately preceding address, the first signal indicates that the address may not be sequential to the immediately preceding address in a same clock cycle as the second sequence signal indicating that the address is not sequential to the immediately preceding address. It should be

noted that the first sequence signal is always provided before the second sequence signal because there must be a hit (first signal) in the BTB before the PC is either incremented or replaced. In addition, it is known that these signals occur in the same cycle so that a predicted address may be outputted for fetching purposes in the very next cycle, thereby keeping the pipeline full. See column 1, lines 38-41. That is, if a branch instruction is fetched in clock cycle X, then a hit will occur and a PC update will be made in that cycle, so that in clock cycle X+1, the instruction from the address pointed to by the new PC may be fetched.

27. Claims 10-12, 18-20, and 26-27 are rejected under 35 U.S.C. 102(e) as being anticipated by Yoshida, U.S. Patent No. 6,205,536. In addition, Hennessy is cited as extrinsic evidence for providing a showing that prediction signals and predicted addresses are provided with each other in the same cycle.

28. Referring to claim 10, Yoshida has taught a processing system for fetching instructions and data; comprising:

a) an address bus for providing a current address for retrieving a first instruction, a previous address for retrieving a second instruction, and a data address for retrieving data, wherein the data address occurs before the current address and after the previous address, and wherein the current address follows the previous address without any intervening addresses for retrieving instructions. See the abstract and Fig.27 and note that the third address provided is a data address, and it occurs between a previous and current address. Note that no intervening instruction addresses exist between instruction addresses 2 and 3.

b) a data bus for retrieving the first and second instructions and the data. This is an inherent component as data and instructions must be transmitted via some medium.

c) a fetch unit, coupled to the address bus and the data bus, for generating a first sequence signal that when asserted for the current address indicates that the current address is sequential to the previous address and when negated indicates that the current address may not be sequential to the previous address, wherein the asserted or negated first sequence signal is provided with the current address. Note from column 10, lines 30-36, that Yoshida has taught branch prediction for branch instructions, which would be a type of instruction represented in Fig.27. Even though Yoshida is silent as to how the branch prediction (first signal) is performed, it is inherent that when predicting branches, they may be predicted taken or not taken. Clearly, if a branch is predicted not taken, then the current address is sequential to the previous address. That is, the system begins fetching from the address immediately following the address of the branch instruction. On the other hand, if the branch is predicted taken, then the current address may not be sequential to the previous address. That is, the system begins fetching from the target address of the branch instruction. However, since the prediction won't be validated or invalidated until the branch is finally executed, the fetching from the target is merely speculative, as opposed to known. Finally, it should be realized that the prediction and the predicted address are determined/provided in the same cycle. This ensures that the pipeline will stay full by allowing a fetch to occur in the cycle just after the previous address. Say, for instance, that in cycle 0, a previous instruction is fetched from a previous address. In that same cycle, a signal specifying whether the branch is predicted taken/not taken will be appropriately set, and the current address will be determined based on that signal. Then in cycle 1, that current address may be fetch from

in cycle 1 (the very next cycle). This ensures that no pipeline disruption occurs. If the signal and current address were not provided in the same cycle, then pipeline disruption would occur and the advantages gained through branch prediction would be at least partially nullified. This concept is further supported by pages 271-273 of Hennessy. That is, if a hit occurs in a branch buffer, then a prediction and a current address (in this case, the predicted address) are provided at the same time. See Fig.4.22 (note that the prediction and address are read from the buffer at the same time in response to a hit of the previous address).

29. Referring to claim 11, Yoshida has taught a system as described in claim 10. Yoshida has further taught an address control unit for receiving a branch condition, a branch decode signal, and a load/store signal, and for providing the first sequence signal. Clearly, it is inherent that each of these signals is present. A branch condition must be realized in order to determine the direction of the branch. In addition, since all instructions are decoded, including branches, a branch decode signal will be provided. And, load/store signals are inherent in any system with registers, which are clearly references in Fig.8-10 (Rn). Any components which receive these signals are part of the address control unit. And, the first sequence signal is also provided by the address control unit.

30. Referring to claim 12, Yoshida has taught a system as described in claim 11. Yoshida has further taught:

a) an execution unit which provides the branch condition. See Fig.7, component 268, and note the condition code. Clearly, if the branch has to reference some condition in order to branch, then that condition must be produced by an execution unit. Usually, the condition is determined through operations such as addition.

b) a decode control unit which provides the branch decode signal and the load/store signal. See Fig.22, component 52, and note that since all instructions are decoded, signals corresponding to the instructions are produced by the decode unit.

31. Referring to claim 18, Yoshida has taught a processing system for fetching instructions and data, comprising:

a) an execution unit. See Fig.22, component 56.

b) a decode control unit. See Fig.22, component 52.

c) a fetch unit, coupled to the execution unit and the decode control unit, for providing instruction and data addresses on an address bus and providing a first sequence signal that indicates whether each instruction address provided on the address bus is sequential to an immediately preceding instruction address even if a data address is provided between the instruction address and the immediately preceding instruction address, wherein, for each instruction address provided on the address bus, the first sequence signal indicating whether the instruction address is sequential to an immediately preceding instruction address is provided with the instruction address. Clearly, since Yoshida's system executes branches (see Fig.7), it must be determined if every instruction address is sequential to its preceding address. That is, every address will either be sequential to its preceding address or it will not be sequential due to a branch. Therefore, the first signal is merely any signal which dictates that a prediction is or is not to be used (and note from column 10, lines 30-36, that prediction is employed). For instance, a first signal exists which would cause the system to either use a predicted address for a branch (which could be non-sequential) or to use an incremented value of the program counter, a component that inherently exists. Furthermore, it can be seen from Fig.27 that data addresses

may occur between instruction addresses. It should be noted that even if a data address occurs between two instruction addresses, the system must still know whether the current instruction address is sequential or not sequential (whether to choose the next or a possible non-sequential prediction) to the previous instruction address, thereby requiring the existence of the first signal. Finally, it should be realized that the prediction and the predicted address are determined in the same cycle. This ensures that the pipeline will stay full by allowing a fetch to occur in the cycle just after the previous address. As a result, say, for instance, that in cycle 0, a previous instruction is fetched from a previous address. In that same cycle, a signal specifying whether the branch is predicted taken/not taken will be appropriately set, and the current address will be determined based on that signal. Then in cycle 1, that current address may be fetch from in cycle 1 (the very next cycle). This ensures that no pipeline disruption occurs. If the signal and current address were not provided in the same cycle, then pipeline disruption would occur and the advantages gained through branch prediction would be at least partially nullified. This is further supported by pages 271-273 of Hennessy. That is, if a hit occurs in a branch buffer, then a prediction and a current address (in this case, the predicted address) are provided at the same time. See Fig.4.22 (note that the prediction and address are read from the buffer at the same time in response to a hit of the previous address).

32. Referring to claim 19, Yoshida has taught a unit as described in claim 18. Yoshida has further taught that the execution unit comprises a condition generator that provides a branch condition signal to the fetch unit. See Fig.7, component 268, and note the condition code. Clearly, if the branch has to reference some condition in order to branch, then that condition

must be produced by an execution unit. Usually, the condition is determined through operations such as addition.

33. Referring to claim 20, Yoshida has taught a unit as described in claim 19. Yoshida has further taught that the decode control unit provides a branch decode signal and a load/store signal to the fetch unit. See Fig.22, component 52, and note that since all instructions are decoded, signals corresponding to the instructions are produced by the decode unit.

34. Referring to claim 26, Yoshida has taught a system as described in claim 10. Yoshida has further taught that the first sequence signal is provided for use by an instruction memory. The prediction (taken/not-taken and predicted address) is used by the instruction memory to provide the target instruction at the predicted address. More specifically, the predicted address is applied to the memory so that an instruction may be fetched.

35. Referring to claim 27, Yoshida has taught a system as described in claim 10. Yoshida has further taught that when the current address is a data address, the first sequence signal is negated. It should be noted that all addresses are data addresses as an address is represented in binary data bits. So when the current address is a data address, the first sequence signal may be negated or asserted. Therefore, claim 27 is anticipated.

Response to Arguments

36. Applicant's arguments filed on August 16, 2005, have been fully considered but they are not persuasive.

37. Applicant argues the novelty/rejection of claim 1 on pages 9-10 of the remarks, in substance that:

> "Firstly, the branch outcome signal alone does not provide the same information as the claimed third sequence signal because it depends on what was predicted originally (by the prediction signal) as to whether or not the branch outcome will indicate that it is not sequential when negated and sequential when asserted."
>
> "Furthermore, the branch outcome signal is not provided with respect to the same current address as the first and sequence signals as claimed in claims 1 and 3.

38. These arguments are not found persuasive for the following reasons:

a) Regarding the first argument, Applicant is arguing limitations that are not in the claim. More specifically, there is no requirement that the third signal provide the claimed information al by itself (without knowing an original prediction. And, applicant's use of the word "comprising" leaves the claim open to having unrecited elements. In Hoyt, the third signal does provide the claimed information based on the prediction.

b) Regarding the second argument, the examiner's interpretation of "current address" is essentially an address from which to fetch an instruction immediately after the previous address. As explained in the examiner's example in the last Office Action, it is possible for two distinct addresses to fall into this category, i.e. two addresses at two different points in time may be the address from which to fetch an instruction immediately after the previous (branch) address. Applicant should not necessarily limit "current address" as being current with respect to time, but instead also leave it open to being current with respect to position. That is, if the branch address (previous address) is 0000 and its target is address is 1111, then if the branch is predicted taken, the current address is 1111, as this is the address to fetch from after the previous address. However, if a misprediction occurs shortly thereafter, then the current address is not 1111, because as a misprediction indicates that 1111 is not the address to fetch from after the previous address. Instead, the current address is 0001, as this is the address to fetch from after the previous address. The examiner agrees with applicant that these addresses are determined at

different times, but again, if applicant leaves "current address" open to interpretation such that an address is current with respect to position, then in either case, both addresses (0001 and 1111) may be considered "the current address" because both contain instructions which, based on the circumstance, are to be fetched immediately after fetching an instruction from the previous address. More specifically, after all is said and done and the branch is resolved, after the previous address, an instruction is fetched from either address 0001 or 1111 (and no other intermediate addresses). In the case where 0001 becomes the current address after an incorrect taken prediction, address 0001 is immediately following the previous address as far as fetching is concerned. Address 1111 was incorrect and therefore, the system is to act as if instructions were never fetched from that location and any subsequent locations. In summation, the claim does not require that "current" be limited to current with respect to time, and therefore, the examiner is interpreting that current may also be current with respect to position (being the next address).

39. Regarding the claim 24 argument, the examiner has responded to the argument in the rejection of claim 24 above. Essentially, Hennessy was supplied as extrinsic evidence showing that the prediction and predicted address are provided in the same cycle.

40. The argument for claim 13 is moot in view of the new grounds of rejection.

41. The arguments for claims 10 and 18 are responded to in the rejections of claims 10 an 18 above. As with claim 10, Hennessy is provided as evidence of showing the providing of the prediction and current address.

42. Regarding the claim 26 argument, a prediction may include a taken/not-taken bit and an address, which is then used by the instruction memory for fetching purposes. See Fig.4.22 of Hennessy.

Conclusion

43. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Applicant is reminded that in amending in response to a rejection of claims, the patentable novelty must be clearly shown in view of the state of the art disclosed by the references cited and the objections made. Applicant must also show how the amendments avoid such references and objections. See 37 CFR § 1.111(c).

Sonobe, U.S. Patent No. 5,826,108, has taught a system having burst mode control in which burst mode may be entered/exited based on a variety of signals.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to David J. Huisman whose telephone number is (571) 272-4168. The examiner can normally be reached on Monday-Friday (8:00-4:30).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

DJH
David J. Huisman
October 3, 2005

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100